

Using Functions in Alice

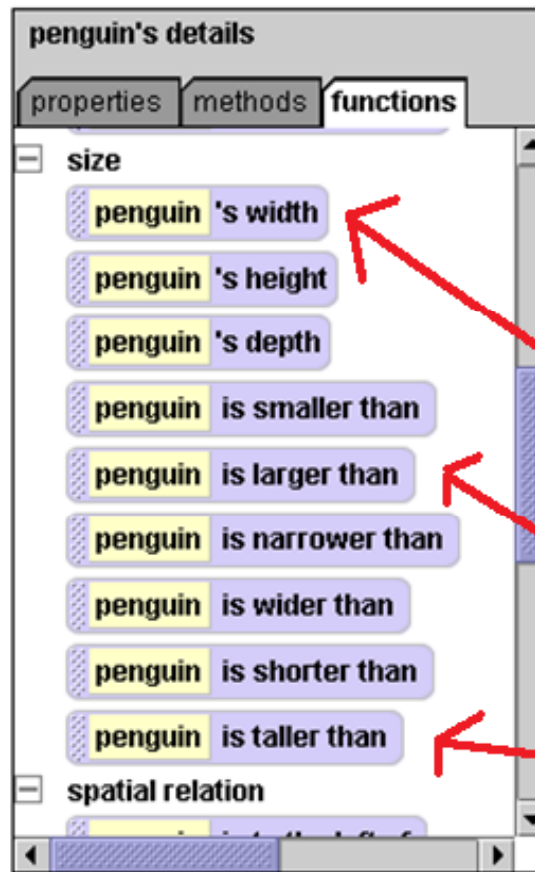
By Jenna Hayes
Under the direction of
Professor Susan Rodger
Duke University
July 2008



Step 1: Understanding Functions

1.

Download the starting world that goes along with this tutorial. We will be using functions. A **function** in Alice is basically a question about information in your Alice world that Alice answers. Click on the **penguin** object in the object tree. Then click on the **functions** tab. You will see a LONG list of functions. Scroll down and look at the functions under **size**.



2.

Each of these functions asks a question about the penguin, and then keeps the answer so that you can use it in your Alice world.

-How wide is the penguin?

-Is the penguin larger than a certain object?

-Is the penguin taller than a certain object?

These functions can be very useful in Alice. What if, for example, you want to make something move up and stand on top of the penguin's head? You don't know how exactly how tall the penguin is. But Alice does!

Step 2: Using the **True** or **False** Functions

Some functions in Alice are statements to which the answer is either **true** or **false**, like the one on the previous page that says, “penguin is taller than.” We want to know whether the penguin is taller than the normalGuy. It’s almost impossible to tell just by looking at them, because their heights are so close together. So we will use a function to know for sure.

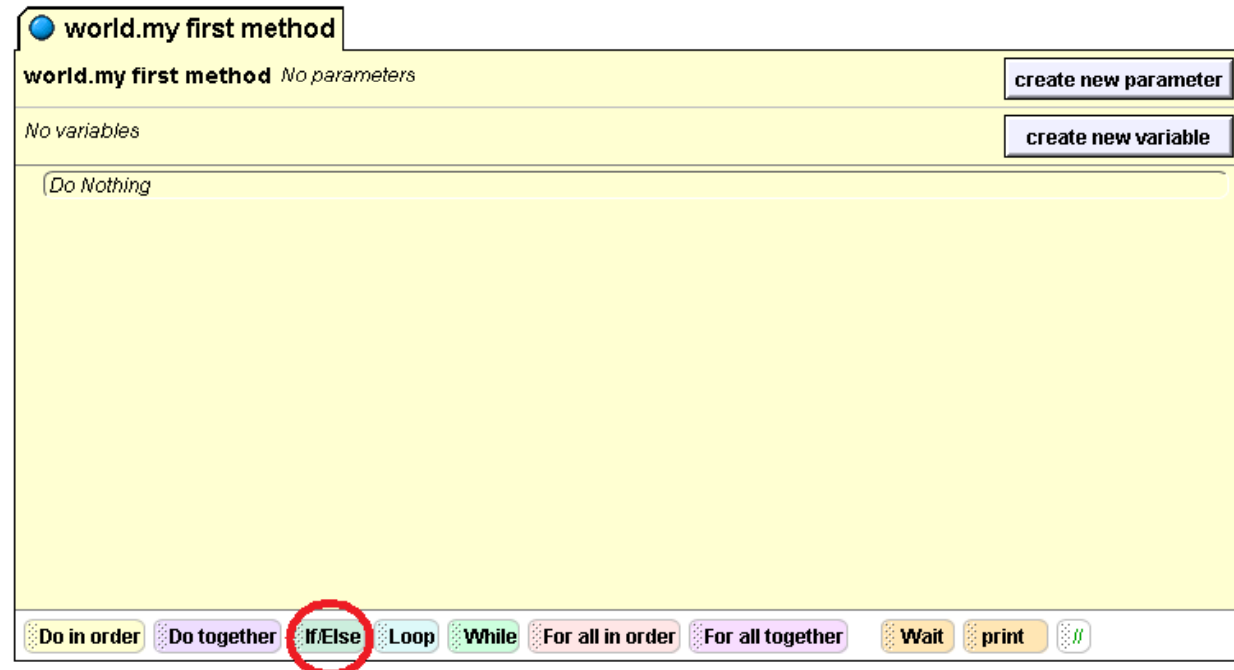
We are going to add commands so that when the Alice world starts, if the penguin is taller, it will say “Hah! I’m taller!”, but if the normalGuy is taller, he will say, “Hah! I’m taller!”.



Step 2: cont.

Since we only want the penguin to say “Hah! I’m taller” IF he is taller, we need to use something called an **If Else** statement. It is located below your method editor:

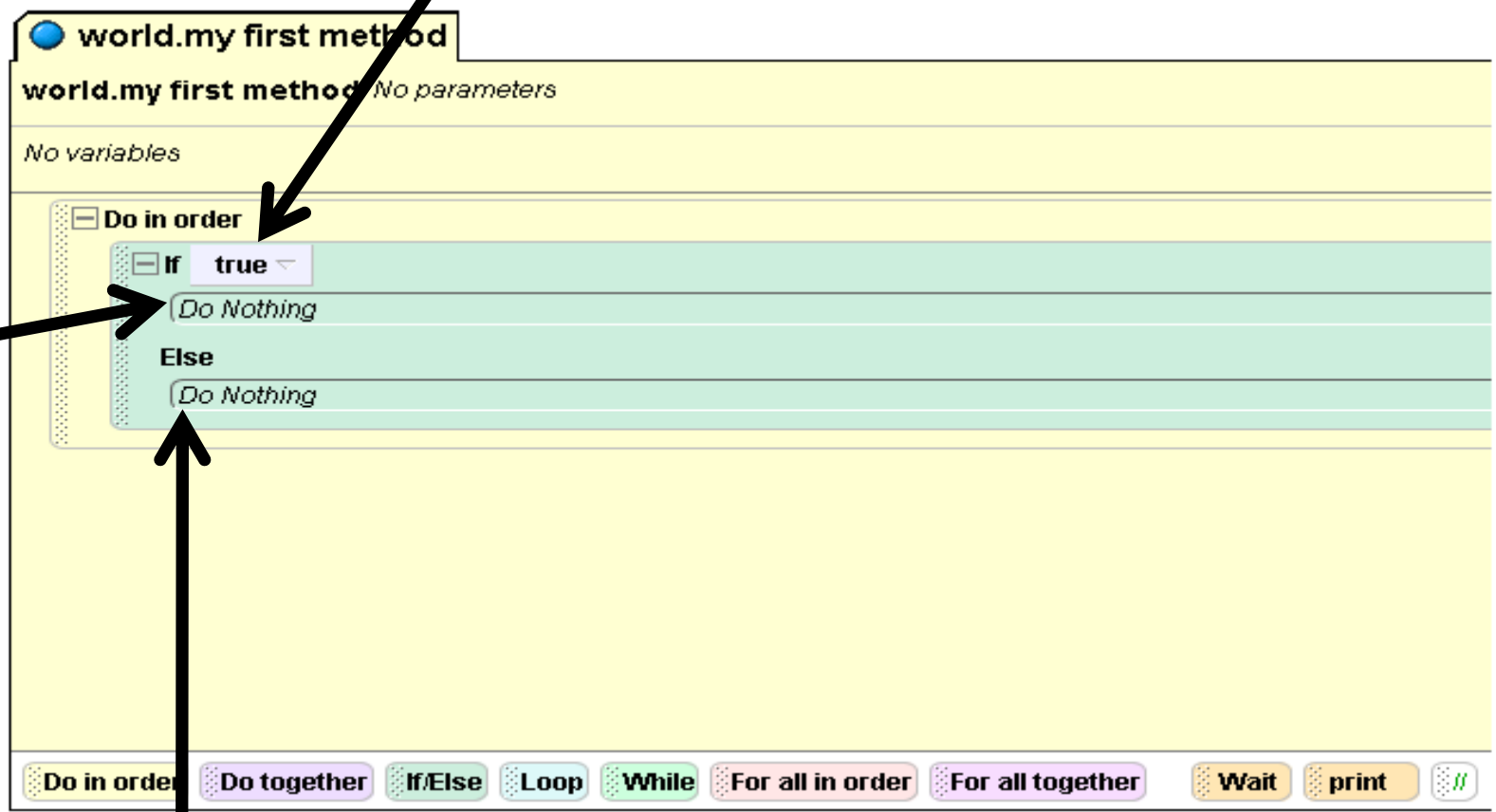
First drag a **Do in order** into your method editor. Then, inside that, drag in an **If Else** statement. Select **true** when you drop it. We will replace this later.



Step 2: cont.

This is where you put whatever you want to happen if the answer to your question is true. This is where we'll tell the penguin to say "Hah! I'm taller!"

Here is where you put the question that is either true or false. For us, that will be **penguin is taller than normalGuy**. Since it is currently set at **true**, that means this If Else says, "If this statement right here is true, do whatever commands are right under it."



This is where you put whatever you want to happen if the answer to your question is NOT true. If it is not true, it will skip everything above the **Else**, and go straight to whatever is here. This is where we'll tell the person to say "Hah! I'm taller!", because if our statement is false, and the penguin is NOT taller, that means the person is taller!

Step 2: cont.

So let's construct our question. Click on **penguin** in the object tree and then click on **functions**. Find the **penguin is taller than** function, and drag it over the word **true** in your **If Else** statement. When the little menu pops up, click on **normalGuy**. Your code will look like this:

The image shows a Scratch code editor window titled "world.my first method". The editor has a yellow background and a toolbar at the bottom with various control blocks: "Do in order", "Do together", "If/Else", "Loop", "While", "For all in order", "For all together", "Wait", "print", and a comment block. The main workspace contains a single "If Else" block. The "If" branch has a condition "penguin is taller than normalGuy" and a "Do Nothing" block. The "Else" branch also has a "Do Nothing" block. The "If Else" block is highlighted with a green border. The "penguin" and "normalGuy" objects are visible in the top right corner of the workspace.

world.my first method

world.my first method No parameters

create new parameter

No variables

create new variable

Do in order

If

penguin is taller than normalGuy

Do Nothing

Else

Do Nothing

Do in order Do together If/Else Loop While For all in order For all together Wait print

Step 2: cont.

Now click on **penguin** in the object tree, and then click on **methods**. Find **penguin say**, and drag it into the method editor under **If**. When the menu pops up, click on **other**, and type in “Hah! I’m taller!”. Your code will look like this:

The image shows the Scratch method editor for a method named "world.my first method". The editor has a yellow background and a title bar with a blue circle icon and the text "world.my first method". Below the title bar, there are two sections: "world.my first method No parameters" and "No variables". To the right of these sections are two buttons: "create new parameter" and "create new variable". The main area of the editor is a light green box with a "Do in order" checkbox. Inside this box, there is an "If" block. The "If" block has a dropdown menu set to "penguin" and a condition "is taller than normalGuy". Below the "If" block is an "Else" block. The "Else" block contains a "Do Nothing" block. At the bottom of the editor, there is a toolbar with various control blocks: "Do in order", "Do together", "If/Else", "Loop", "While", "For all in order", "For all together", "Wait", "print", and a "Run" button with a green flag icon.

world.my first method

world.my first method No parameters

No variables

☐ Do in order

☐ If **penguin** is taller than **normalGuy**

penguin say **Hah! I'm taller!** more...

Else

Do Nothing

Do in order Do together If/Else Loop While For all in order For all together Wait print Run

Step 2:

Now click on **normalGuy** in the object tree, and click on **methods**. Find **normalGuy say** and put under the **Else** part of your If Else statement. Tell him to say “Hah! I’m taller!” Your code will look like this:

The image shows a Scratch code editor window titled "world.my first method". The method has no parameters and no variables. The code is as follows:

```
Do in order
  If
    penguin is taller than normalGuy
      penguin say Hah! I'm taller! more...
    Else
      normalGuy say Hah! I'm taller! more...
```

The code is written in a light blue box. The "If" block is a light blue block with a "penguin" object and an "is taller than" comparison. The "Else" block is a light blue block with a "normalGuy" object. Both blocks have a "say" block with the text "Hah! I'm taller!" and a "more..." block. The "Do in order" checkbox is checked. The bottom of the editor shows a palette with various control blocks: "Do in order", "Do together", "If/Else", "Loop", "While", "For all in order", "For all together", "Wait", "print", and a comment block.

Now play your world. Who is taller, the penguin or the person?

Step 3: Using Number Functions

Now we're going to use one of the functions that is a question whose answer is a number. We're going to make the penguin move right up to the man and give him a hug. The only problem is, we don't know how far to tell him to move! That's why we'll use a function!

First we'll tell the man to say something. He's tired of competing with the penguin about their height, and he wants to be friends. Click on **normalGuy** in the object tree, then click on **methods**. Find **normalGuy say**, and drag it under your If Else statement. Make him say "Let's be friends. Give me a hug, penguin!" Your code will look like this:

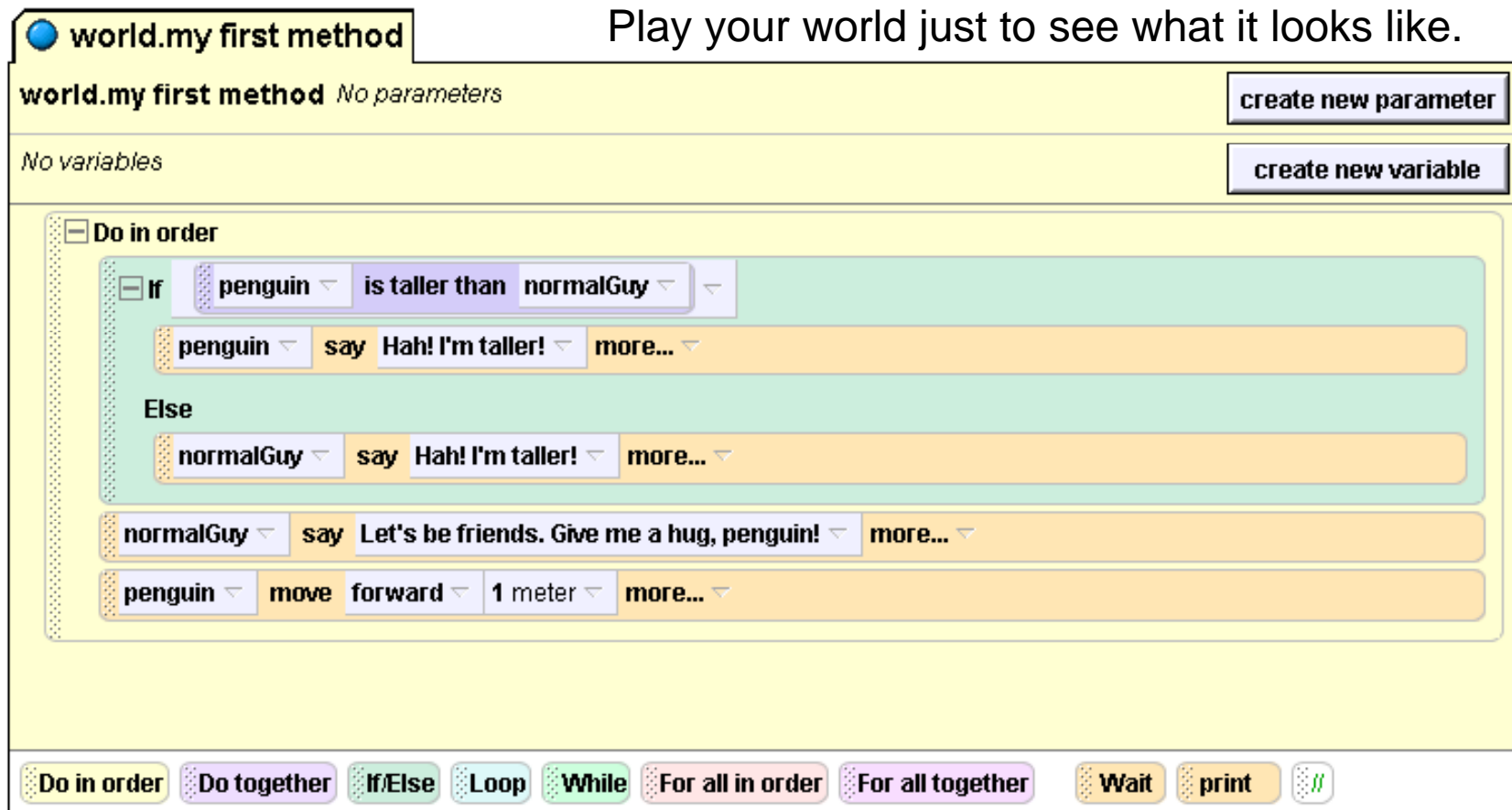
The image shows a Scratch code editor window titled "world.my first method". The script area contains the following code:

```
world.my first method No parameters  
No variables  
Do in order  
  If penguin is taller than normalGuy  
    penguin say Hah! I'm taller! more...  
  Else  
    normalGuy say Hah! I'm taller! more...  
    normalGuy say Let's be friends. Give me a hug, penguin! more...
```

At the bottom of the editor, there is a palette of control blocks: "Do in order", "Do together", "If/Else", "Loop", "While", "For all in order", "For all together", "Wait", "print", and a comment block.

Step 3: cont.

Now we want the penguin to move right up to the guy to hug him. Click on **penguin** in your object tree and then click on **methods**. Find the **penguin move** and drag it into your method editor under everything else. We don't know exactly how far yet, so just put 1 meter for now. Your code will look like this:

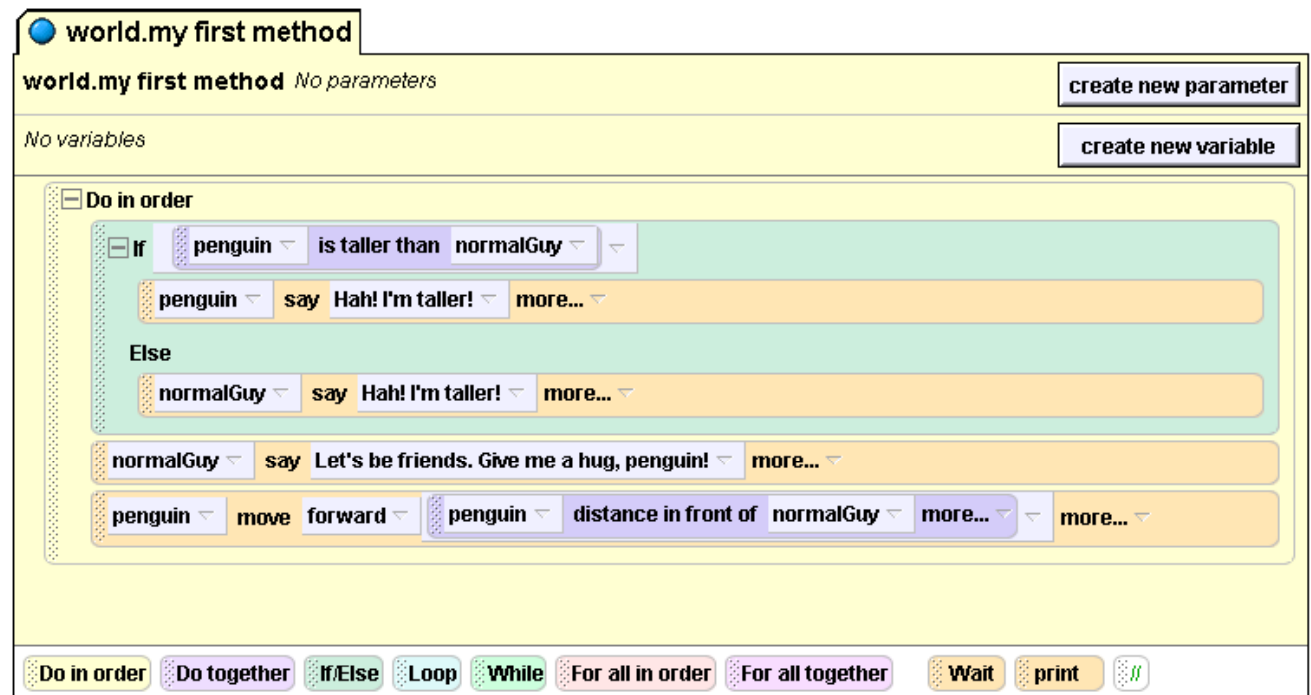
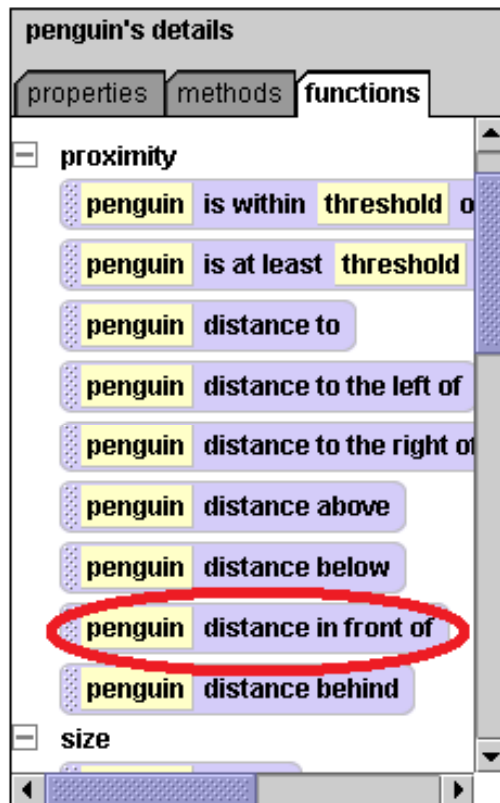


The image shows a Scratch world editor interface. At the top, there's a tab labeled "world.my first method" with a blue circle icon. To its right, the text "Play your world just to see what it looks like." is displayed. Below the tab, the main area is yellow and contains the text "world.my first method No parameters" and "No variables". On the right side of this yellow area, there are two buttons: "create new parameter" and "create new variable". The main workspace is a large yellow rectangle. Inside it, there's a "Do in order" block (indicated by a minus sign and the text "Do in order"). This block contains several sub-blocks: an "if" block with the condition "penguin is taller than normalGuy", followed by a "say" block for "penguin" with the text "Hah! I'm taller!" and a "more..." dropdown. Below the "if" block is an "Else" block containing a "say" block for "normalGuy" with the text "Hah! I'm taller!" and a "more..." dropdown. Below the "Else" block is a "say" block for "normalGuy" with the text "Let's be friends. Give me a hug, penguin!" and a "more..." dropdown. At the bottom of the "Do in order" block is a "move" block for "penguin" with the text "forward 1 meter" and a "more..." dropdown. At the bottom of the interface, there's a row of buttons: "Do in order", "Do together", "If/Else", "Loop", "While", "For all in order", "For all together", "Wait", "print", and a comment icon (two parallel slanted lines).

Step 3: cont.

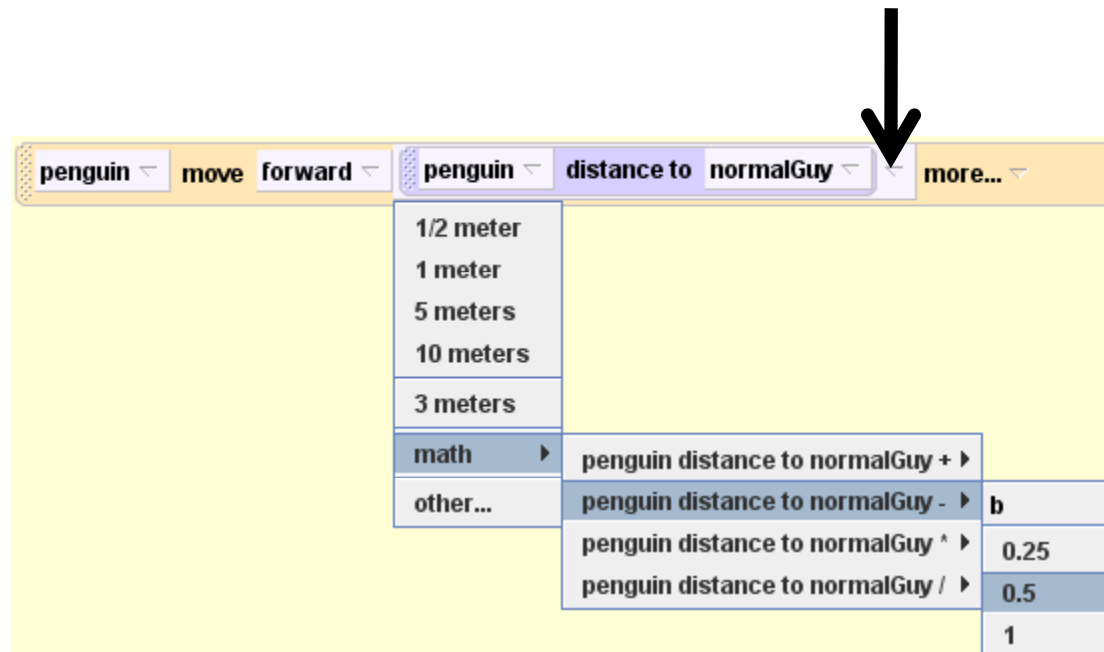
Now we're going to use a function to tell the penguin how far forward to move. Click on **penguin** in your object tree, and then click on the **functions** tab. Find the function called **distance in front of**.

Drag that function over the **1 meter** on your **penguin move forward** command and drop it there. On the menu that pops up, select **normalGuy**. Your code will look like this:



Step 3: cont.

Try playing your world. What happens? The penguin moves too far, into the body of the normalGuy. It would be nice if the penguin would stop about 0.5m before normalGuy. We can select math, followed by “-”, followed by a number. Click here to apply math.

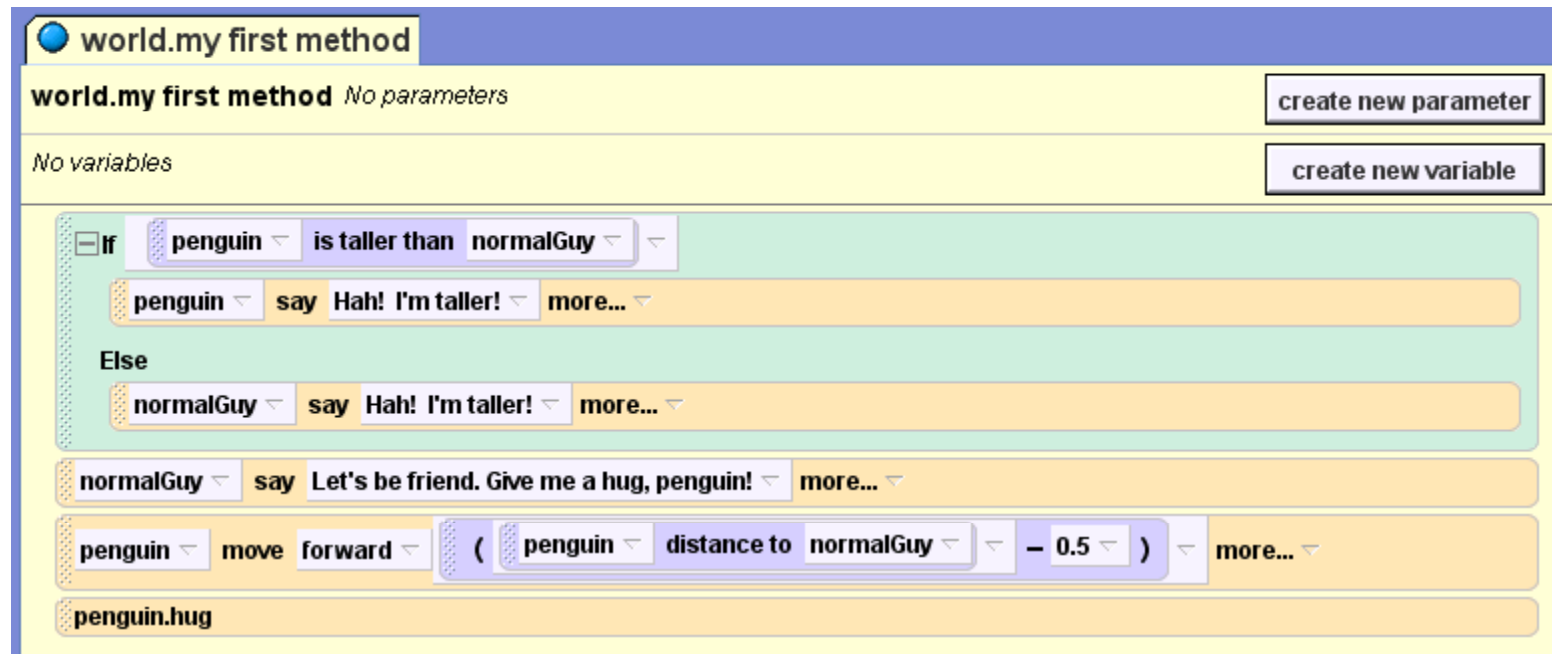


Your code for this line will then look like this:



Step 4: Finishing Up

The last thing you need to do is make the penguin hug the guy. Click on **penguin** in the object tree, and then click on the **methods** tab. Find the **hug** method (near the top of the methods) and drag it into your method editor under everything else. Your final code will look like this:



Now play your world and watch the magic!

Now you know the basics of using functions. We only used two functions in this tutorial, but there are MANY functions in the Alice world that can be useful. Feel free to explore the uses of other functions.

